

CLAIMS

What is claimed is:

1. A state machine for an assembler capable of processing structured assembly
2 language, said state machine comprising:

3 an IF state, an ELSE state, an END_IF state, an ELSE_IF state, and a
SETUP_IF state;

means for transitioning from said IF state or said ELSE_IF state to said
SETUP_IF state, in response to recognizing a SETUP_IF clause; and

means for transitioning from said SETUP_IF state to said ELSE_IF state,
in response to recognizing an ELSE_IF clause.

1 2. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said IF state to said ELSE state, in response to recognizing an ELSE
3 clause.

1 3. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said IF state to said END_IF state, in response to recognizing an
3 END_IF statement.

1 4. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said IF state to said ELSE_IF state, in response to recognizing an
3 ELSE_IF clause.

1 5. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said ELSE state to said END_IF state, in response to recognizing an
3 END_IF statement.

1 6. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said ELSE_IF state to said END_IF state, in response to recognizing
3 an END_IF statement.

1 7. The state machine of Claim 1, wherein said state machine further includes a means
2 for transitioning from said ELSE_IF state to said ELSE state, in response to recognizing
3 an ELSE clause.

1 8. A computer program product residing on a computer usable medium for processing
2 structured assembly language, said computer program product comprising:

3 program code means for implementing a state machine having an IF state,
4 an ELSE state, an END_IF state, an ELSE_IF state, and a SETUP_IF state;

5 program code means for transitioning from said IF state or said ELSE_IF
6 state to said SETUP_IF state, in response to recognizing a SETUP_IF clause; and

7 program code means for transitioning from said SETUP_IF state to said
8 ELSE_IF state, in response to recognizing an ELSE_IF clause.

PROVISIONAL EXAMINER'S DRAFT

1 9. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said IF state to said ELSE state,
3 in response to recognizing an ELSE clause.

1 10. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said IF state to said END_IF
3 state, in response to recognizing an END_IF statement.

1 11. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said IF state to said ELSE_IF
3 state, in response to recognizing an ELSE_IF clause.

1 12. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said ELSE state to said
3 END_IF state, in response to recognizing an END_IF statement.

1 13. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said ELSE_IF state to said
3 END_IF state, in response to recognizing an END_IF statement.

1 14. The computer program product of Claim 8, wherein said computer program product
2 further includes program code means for transitioning from said ELSE_IF state to said
3 ELSE state, in response to recognizing an ELSE clause.

1 15. A data processing system having an assembler for processing structured assembly
2 language, said data processing system comprising:

3 a state machine having an IF state, an ELSE state, an END_IF state, an
4 ELSE_IF state, and a SETUP_IF state;

5 means for transitioning from said IF state or said ELSE_IF state to said
6 SETUP_IF state, in response to recognizing a SETUP_IF clause; and

 means for transitioning from said SETUP_IF state to said ELSE_IF state,
 in response to recognizing an ELSE_IF clause.

1 16. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said IF state to said ELSE state, in response
3 to recognizing an ELSE clause.

1 17. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said IF state to said END_IF state, in response
3 to recognizing an END_IF statement.

1 18. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said IF state to said ELSE_IF state, in
3 response to recognizing an ELSE_IF clause.

1 19. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said ELSE state to said END_IF state, in
3 response to recognizing an END_IF statement.

1 20. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said ELSE_IF state to said END_IF state, in
3 response to recognizing an END_IF statement.

1 21. The data processing system of Claim 15, wherein said data processing system
2 further includes means for transitioning from said ELSE_IF state to said ELSE state, in
3 response to recognizing an ELSE clause.

1 22. A data processing system comprising:

2 means for identifying a SETUP_IF clause;

3 means for associating said identified SETUP_IF clause with an ELSE_IF
4 clause having a test condition; and

5 means for inserting instructions from said identified SETUP_IF clause prior
to the test condition of said ELSE_IF clause where said ELSE_IF clause logically
follows a prior IF clause or a prior ELSE_IF clause.

DRAFT